IT2053 – Rapid Application Development

ASSIGNMENT II

Learning Outcomes:

Use C# technologies for developing desktop applications. Select and apply technologies for building desktop application.

Instructions:

- Answer ALL questions.
- Marks will be awarded for good presentation and thoroughness in your approach.
- **NO** marks will be awarded for the entire assignment if any part of it is found to be copied directly from printed material or from another student.
- Complete this cover sheet and attach it to your assignment.

Student Declaration:

I declare that:

- *I understand what is meant by plagiarism.*
- *The implications of plagiarism have been explained to me by my institution.*
- *This assignment is all my own work and I have acknowledged any use of the published or unpublished works of other people.*

Student's signature: Date:

Total number of pages including cover page			11 pages		
Submission Date	2023-02-20	**Due Date**		**Class Code**	

Student Name	**Index No.**
KHMA DILSHAN	D/BIT/22/0031
JMDR JAYANATH	D/BIT/22/0032
DTN WIJESURIYA	D/BIT/22/0002
PPK WICKRAMAGE	D/BIT/22/0012

Lecture's Name	Ms. D Gunasekara

OFFICIAL USE ONLY

MARKER'S COMMENTS	
Marker's Name	**Marks Awarded (100%)**

Software Requirements Specification

Version 1.0
<<Annotated Version>>

February 20, 2023

KDU Club Management System

KHMA Dilshan
JMDR Jayanath
DTN Wijesuriya
PPK Wickramage

Submitted in partial fulfillment of the requirements of
IT 2053 Rapid Application Development

Table of Contents

Table of Contents i
List of Figures i
1.0. Introduction 1
1.1. Purpose 1
1.2. Scope of Project 1
2.0. Overall Description 2
2.1. System Environment 2
2.2. Functional Requirements Specification 3
2.2.1. Club Member Use Case 3
2.2.1.a. Use case: Join with Club 3
2.2.1.b. Use case: Participate Online Meetings / Events 4
2.2.1.c. Use case: Pay Membership Fees and Club Related Payments 4
2.2.2. CMS Admin Use Cases 5
Update / Delete / Edit - Meetings / Events use case: 5
Check Status & Maintain Regulations use case: 6
2.3. Non-Functional Requirements Specification 6
2.4. User Characteristics 7
2.5. Non-Functional Requirements 7
3.0. Requirements Specification 7
3.0.1. Functional Requirements 7
3.0.1.a. Join the Club 7
3.0.1.b. Participate Online Meetings / Events 8
3.0.1.c. Pay Membership Fees and Club Related Payments 8
3.0.1.d. Update / Delete / Edit - Meetings / Events 8
3.0.1.e. Check Status & Maintain Regulations 8

List of Figures

Figure 1 - System Environment 2
Figure 2 - CMS Admin Use Cases 5

<< Any comments inside double brackets such as these are *not* part of this Software Requirement Specification (SRS) but are comments upon this SRS example to help the reader understand the point being made. >>

1.0. Introduction

1.1. Purpose

This document is for the Software Requirements Specification (SRS) of the KDU Club Management System (KDU-CMS). It contains detailed functional, non-functional, and requirements baseline for the development of the system. The SRS serves as the official means of communicating user requirements to the developer and provides a common reference point for both the developer team and stakeholders. The SRS will evolve over time as users and developers work together to validate, clarify, and expand its contents.

1.2. Scope of Project

Sir General John Kotelawala Defence University have many kinds of degree programs. All the undergraduates are divided to faculties and departments under their related degree program. But KDU Clubs are open to all the undergraduates. So, if you are a fresher to university, how you know what the clubs are in there, club details, and membership fees or you are already undergraduate of university, informed about future events and meetings on clubs. These problems are solved with this system. This project is work like as an information portal. We are hoping this project will definitely help to manage the Clubs of KDU. The purpose of the KDU Club Management System (KDU-CMS) is to create a convenient and easy-to-use application for club members and club seekers of KDU undergraduates who are trying to engage with club activities. The system is based on a relational database with its member management and other usual club related functions. Above all, we hope to provide a comfortable user experience.

2.0. Overall Description

2.1. System Environment

This is a specialized software application that is designed to automate various administrative tasks, such as membership management, event scheduling, membership fees and other payments, and more. Overall, the system environment in club management can be complex, depending on the specific needs of the club and its members. A well-designed and integrated system environment can help clubs to streamline their operations, improve member engagement, and ultimately, achieve their goals more efficiently.

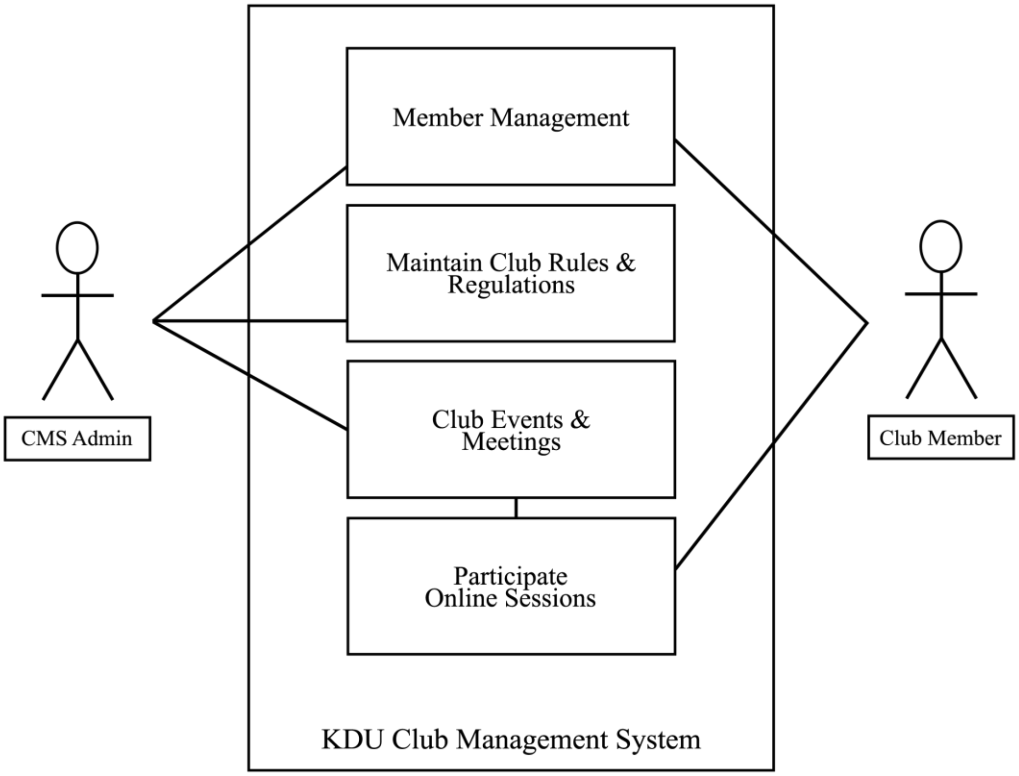

Member Management
Maintain Club Rules & Regulations
Club Events & Meetings
Participate Online Sessions
KDU Club Management System
CMS Admin
Club Member

Figure 1 - System Environment

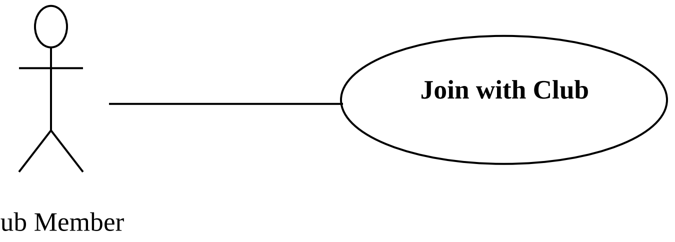
KDU Club Management System has two active actors and one co-operating system. The CMS Admin have access to member management, maintain club rules and regulations, and publish club events and meetings. Club members have access to create their own profile and get the news about sessions.

<< In this Figure 1 has demonstrate how a club will work, The KDU-CMS have many clubs. So, all the clubs are operating literally like same way. Each club have at least one of Admin. >>

2.2. Functional Requirements Specification

2.2.1. Club Member Use Case

2.2.1.a. Use case: Join with Club
Diagram:



Brief Description
The Club Member has access to create their own profile and after profile creation, they can join with their interested clubs.

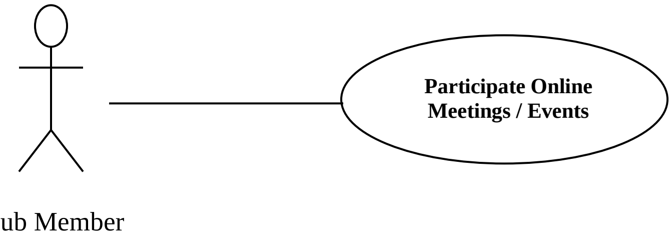
Initial Step-By-Step Description
Before this use case can be initiated, the member has already accessed the own profile creation.

1. The Member needs to create their own profile entering Name, Index Number, KDU Mail Address, Contact Number, Intake, and Department Name.
2. Member details save in the database.
3. The system is automatically sending profile verification email that entered mail address.
4. After verification the system displays the member eligible to join with clubs.
5. The Member needs to just click on club logo and after entering inside the club page, click join button to join with club.

<< Once a member register with the system, that details are save in database and no need to enter again and again, also details can update anytime. >>

2.2.1.b. Use case: Participate Online Meetings / Events

Diagram:



Brief Description

The Club Member has chance to participate online meetings / events through the CMS system.

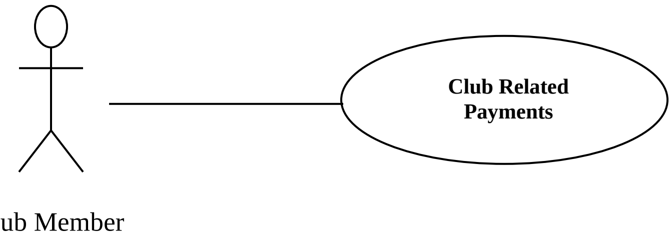
Initial Step-By-Step Description

Before this use case can be initiated, the member has already accessed join with club.

1. Before the participation, the member needs to register with online meeting or event.
2. The system displays all the meetings / events in page, and they are categorized as a club based.
3. The Member needs to choose the meeting or event and click register button.
4. The system is recording your registrations and add to calendar.
5. Additionally, the system will remind time & date of meetings / events held on.

2.2.1.c. Use case: Pay Membership Fees and Club Related Payments

Diagram:



Brief Description

The Club Member has access to pay membership fees and club related payments.

Initial Step-By-Step Description

Before this use case can be initiated, the member has already accessed join with club.

1. Membership Fees and Club related Payments are different in each club, it is published by CMS Admin of each club.
2. The system displays payment charges related with their club.
3. The Member needs to pay that fee using system payment gateway.
4. The system is recording your proceedings and show that report to CMS Admin.

2.2.2. CMS Admin Use Cases
The CMS Admin has the following sets of use cases:

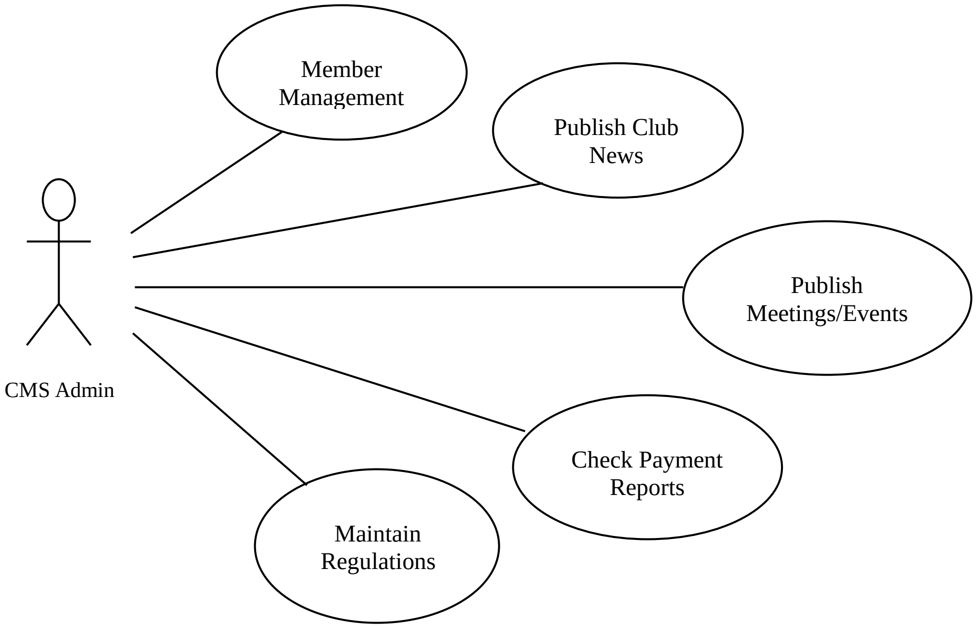
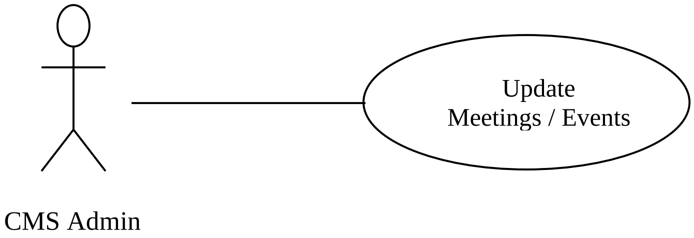


Figure 2 - CMS Admin Use Cases

Update / Delete / Edit - Meetings / Events use case

Use case: Update / Delete / Edit - Meetings / Events
Diagram:



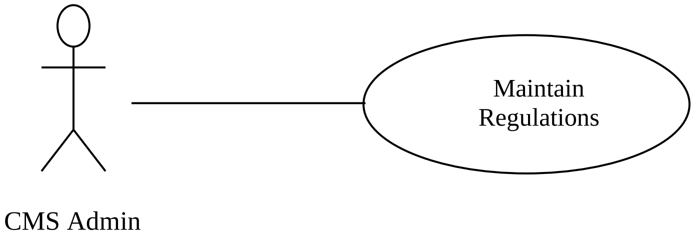
Brief Description
The CMS Admin enters information about future meetings.

Initial Step-By-Step Description
Before this use case can be initiated, the CMS Admin has already published meeting details.

1. The CMS Admin needs to select meeting or event.
2. The system presents the information about the chosen meeting.
3. The CMS Admin Update / Delete / Edit and submits the details.
4. The system verifies the information and returns to the admin dashboard.

Check Status & Maintain Regulations use case

Use case: Check Status & Maintain Regulations
Diagram:



Brief Description
The CMS Admin checks the status of all active members in club and their adaption of regulations.

Initial Step-By-Step Description
Before this use case can be initiated, the CMS Admin has already published rules and regulations of club.

1. The CMS Admin needs to go to admin dashboard.
2. The system presents the information about the members like who are active members on system.

2.3. Non-Functional Requirements Specification

- Performance: The system should be able to handle a large number of users and transactions without significant delays or crashes. Response time should be fast, and the system should be able to handle a high volume of data without slowing down.
- Security: The system should be secure and protect user data. User authentication and access controls should be implemented to prevent unauthorized access, and data encryption should be used to protect sensitive data.
- Usability: The system should be easy to use and navigate, with clear and intuitive user interfaces. Users should be able to find the information they need quickly and easily.
- Scalability: The system should be able to scale to accommodate future growth in the number of users and data volume without requiring major changes to the underlying infrastructure.
- Availability: The system should be available 24/7 with minimal downtime for maintenance and upgrades. It should be able to recover quickly from any failure or downtime.
- Reliability: The system should be reliable and accurate, with consistent performance and data accuracy. The system should also be able to recover from errors or failures without significant impact on users.
- Maintainability: The system should be easy to maintain and upgrade, with clear documentation and support for troubleshooting and bug fixing.

2.4. User Characteristics

The Club Member is expected to get all joined club news in through the single system and be able to use as an information portal. The main screen of the KDU CMS will have all the KDU Clubs and other club related posts.

The CMS Admin is expected it would be a nice interfaced dashboard to handle the club members and get reports.

2.5. Non-Functional Requirements

The system database will be on a server with high-speed Internet capability. The physical machines to be used will be installation for software application. The speed of the Member's connection will depend on the hardware used rather than characteristics of this system.

System is open source and free for use, but it would be allowed for KDU Undergraduates only. The System checks and maintain security using entered details like index number, KDU mail address and also name match to existing undergraduate from university. It prevents outside persons logging and interact with system.

3.0. Requirements Specification

3.0.1. Functional Requirements

3.0.1.a. Join the Club

Use Case Name	Join the Club
Trigger	Buildup community skills
Precondition	Create their own profile
Basic Path	1. The Member needs to create their own profile entering Name, Index Number, KDU Mail Address, Contact Number, Intake, and Department Name. 2. The system is automatically sending profile verification email that entered mail address. 3. After verification the system displays the member eligible to join with clubs. 4. The Member needs to just click on club logo and after entering inside the club page, click join button to join with club.
Other	None

3.0.1.b. Participate Online Meetings / Events

Use Case Name	Participate Online Meetings / Events
Trigger	Undergraduates need to get career and degree related knowledge from professionals through the club sessions
Precondition	Join with the club
Basic Path	1. Before the participation, the member needs to register with online meeting or event. 2. The system displays all the meetings / events in page, and they are categorized as a club based. 3. The Member needs to choose the meeting or event and click register button. 4. The system is recording your registrations and add to calendar.
Postcondition	The system will remind time & date of meetings / events held on
Exception Paths	Somehow, registration process was failed, the system displays try again with registration.

3.0.1.c. Pay Membership Fees and Club Related Payments

Use Case Name	Pay Membership Fees and Club Related Payments
Trigger	Members need to pay membership fees in annually
Precondition	Join with the club
Basic Path	1. Membership Fees and Club related Payments are different in each club, it is published by CMS Admin of each club. 2. The system displays payment charges related with their club. 3. The Member needs to pay that fee using payment gateway.
Postcondition	The system is recording your proceedings and show that report to CMS Admin.

3.0.1.d. Update / Delete / Edit - Meetings / Events

Use Case Name	Update / Delete / Edit - Meetings / Events
Trigger	Update the community about what's going on
Precondition	The CMS Admin needs to publish meeting details
Basic Path	1. The CMS Admin needs to select meeting or event. 2. The system presents the information about the meeting. 3. The CMS Admin Update / Edit and submits the details.
Postcondition	The system verifies the information and returns to the admin dashboard.

3.0.1.e. Check Status & Maintain Regulations

Use Case Name	Check Status & Maintain Regulations
Trigger	Get report about active participation of members
Precondition	The CMS Admin needs to publish rules and regulations
Basic Path	1. The CMS Admin needs to go to admin dashboard. 2. The system presents the information about the members like who are active members on system.